We claim:

1. A method, comprising:
providing at least three elements, including a first element and a last element, each element having an associated parameter;
providing a first identifier for the first element;
for a first sequential execution of the at least three elements, performing a first operation on the first identifier and at least one of the parameters to produce a transform;
saving the transform; and
for a second sequential execution of the elements, performing a second operation on the transform to produce a last identifier associated with the last element.

2. The method of claim 1, wherein performing the first operation includes:
performing a first logic function on the first identifier and the parameter associated with the first element to produce a next identifier for a next one of the elements;
repeating the performing a first logic function on the next identifier and the parameter associated with the next one of the elements, until a last identifier has been produced that is associated with the last element; and
performing a second logic function on the first identifier and the last identifier to produce the transform.

3. The method of claim 2, wherein performing the first logic function includes:
shifting the first identifier to produce a shifted identifier; and
performing an exclusive OR operation on the shifted identifier and the associated parameter to produce the next identifier.

4. The method of claim 3, wherein shifting the first identifier includes shifting the first identifier one bit to the left relative to the associated parameter.

5. The method of claim 2, wherein performing the second logic function includes:
shifting the first identifier to produce a shifted identifier; and
performing an exclusive OR operation on the shifted identifier and the last identifier to produce the transform.

6. The method of claim 5, wherein shifting the first identifier includes shifting the first identifier to the left relative to the last identifier by a number of bits equal in value to one less than a quantity of the at least three elements.

7. The method of claim 1, wherein performing the second operation includes:
shifting the first identifier to produce a shifted identifier; and
performing an exclusive OR operation on the shifted identifier and the transform to produce the last identifier.

8. The method of claim 7, wherein shifting the first identifier includes shifting the first identifier to the left relative to the transform by a number of bits equal in value to one less than a quantity of the at least three elements.

9. The method of claim 1, wherein the at least three elements are branch instructions in an instruction execution pipeline.

10. The method of claim 1, further comprising:
using the last index to access a location in a prediction array; and
using a content of said location to predict a decision status of the last element.

11. A circuit, comprising:

first and second exclusive OR circuits;

a first register having an output coupled to a first input of the first exclusive OR circuit and having an input coupled to an output of the first exclusive OR circuit;

a second register having an output coupled to a first input of the second exclusive OR circuit and having an input;

a multiplexer having an output coupled to the input of the second register, having a first input coupled to the output of the first exclusive OR circuit, and having a second input coupled to the output of the second register;

wherein the output of the first exclusive OR circuit is coupled to a second input of the second exclusive OR circuit.

12. The circuit of claim 11, wherein the first input of the first exclusive OR circuit is offset from the output of the first register to perform a bit-shift operation on data received from the first register by the first exclusive OR circuit.

13. The circuit of claim 11, wherein the second input of the multiplexer is offset from the output of the second register to perform a bit-shift operation on data received from the second register by the multiplexer.

14. The circuit of claim 11, further comprising an array coupled to an output of the second exclusive OR circuit to store transforms generated by the second exclusive OR circuit.

15. The circuit of claim 14, further comprising a counter to generate a shift value to store in the array.

16. A computer system, comprising:

an instruction execution pipeline; and

a branch prediction circuit coupled to the instruction execution pipeline and including:

first and second exclusive OR circuits;

a first register having an output coupled to a first input of the first exclusive OR circuit and having an input coupled to an output of the first exclusive OR circuit;

a second register having an output coupled to a first input of the second exclusive OR circuit and having an input; and

a multiplexer having an output coupled to the input of the second register, having a first input coupled to the output of the first exclusive OR circuit, and having a second input coupled to the output of the second register;

wherein the output of the first exclusive OR circuit is coupled to a second input of the second exclusive OR circuit.

17. The computer system of claim 16, wherein the first input of the first exclusive OR circuit is offset from the output of the first register to perform a bit-shift operation on data received from the first register by the first exclusive OR circuit.

18. The computer system of claim 16, wherein the second input of the multiplexer is offset from the output of the second register to perform a bit-shift operation on data received from the second register by the multiplexer.

19. The computer system of claim 16, further comprising an array coupled to an output of the second exclusive OR circuit to store transforms generated by the second exclusive OR circuit.

20. The computer system of claim 19, further comprising a counter to generate a shift value to store in the array.

21. A circuit, comprising:
a register;
a data shifting circuit having an input coupled to an output of the register;
an exclusive OR circuit having a first input coupled to an output of the data shifting circuit;
an array coupled to a second input of the exclusive OR circuit to transfer transform data to the exclusive OR circuit, and further coupled to the data shifting circuit to transfer data shift information to the data shifting circuit; and
a prediction logic circuit coupled to an output of the exclusive OR circuit.

22. The circuit of claim 21, further comprising a multiplexer having:
an output coupled to an input of the register;
a first input to receive initial data; and
a second input coupled to the output of the exclusive OR circuit.

23. The circuit of claim 21, wherein the data shifting circuit includes a plurality of inputs coupled to the output of the register to shift data from the register by a selected number of bits.

24. A computer system comprising:
an instruction execution pipeline;
a transform generation circuit coupled to the instruction execution pipeline and including:
a register;
a data shifting circuit having an input coupled to an output of the register;

an exclusive OR circuit having a first input coupled to an output of the data shifting circuit;

an array coupled to a second input of the exclusive OR circuit to transfer transform data to the exclusive OR circuit, and further coupled to the data shifting circuit to transfer data shift information to the data shifting circuit; and

a prediction logic circuit coupled to an output of the exclusive OR circuit.

25. The computer system of claim 24, further comprising a multiplexer having:

an output coupled to an input of the register;

a first input to receive initial data; and

a second input coupled to the output of the exclusive OR circuit.

26. The computer system of claim 24, wherein the data shifting circuit includes a plurality of inputs coupled to the output of the register to shift data from the register by a selected number of bits.

27. A machine-readable medium having stored thereon instructions, which when executed by at least one processor cause said at least one processor to perform:

providing at least three elements, including a first element and a last element, each element having an associated parameter;

providing a first identifier for the first element;

for a first sequential execution of the at least three elements, performing a first operation on the first identifier and at least one of the parameters to produce a transform;

saving the transform;

for a second sequential execution of the elements, performing a second operation on the transform to produce a last identifier associated with the last element;

using the last identifier to access a location in a prediction array; and

using a content of said location to predict a decision status of the last element.